

REMARKS

This paper is responsive to a Non-Final Office action dated May 5, 2006. Claims 1-43 were examined.

Claim Objections

Claims 27, 28, 30, and 31 are objected to because of informalities. Accordingly, Applicant has amended the claims to correct punctuation.

Claim Rejections – 35 U.S.C. §112

Claims 27, 28, 30 and 31 have been rejected under 35 U.S.C. §112, for antecedent basis error. Accordingly, Applicant has amended claims 27 and 28 to recite “access” instead of “pop,” and amended claims 30 and 31 to recite “access” instead of “push.”

Claim Rejections – 35 U.S.C. §101

Claims 40-42 are rejected under 35 U.S.C. §101 as being directed to non-statutory subject matter. Apparently, claim 40 is rejected as being directed to non-statutory subject matter, because claim 42 includes wires. Applicant respectfully notes that claim 42 actually recites “wireline, wireless” not wires. Furthermore, the Examiner does not identify anything within claim 40 to support the Examiner’s conclusion regarding statutory subject matter. No reason is given for the rejection of claim 41. Despite skepticism regarding Office policy, Applicant has amended claim 42 to excise recitation of “wireline, wireless.”

Applicant respectfully submits that claims 40 – 42 are directed to statutory subject matter. The Interim Guidelines for Examination of Patent Applications for Patent Matter Subject Matter Eligibility clearly indicate that a computer program encoded in a machine-readable medium is statutory subject matter. See Annex IV, section a.

The Examiner adds a commentary that a computer program product encoded in a computer readable medium includes a program written on a piece of paper. Applicant respectfully requests identification of a computer that can execute a program written on a piece

of paper. If a computer can read a program from a piece of paper and execute the program, then a piece of paper qualifies as a computer readable medium.

Claim Rejections – 35 U.S.C. §102

Claims 1-2, 4-5, 15-19, 25-28, and 30-31 are rejected under 35 U.S.C. §102(b) as being taught by Janice M. Stone's "A simple and correct shared-queue algorithm using Compare-and-Swap", 1990 (hereinafter, "Stone").

The Examiner's understanding of Stone is erroneous and it is abundantly clear that Stone does not disclose or suggest any of the rejected claims. In Stone, if a CSDBL fails, then the procedure repeats execution of the CSDBL. Stone does not disclose or suggest being able to detect a boundary condition (i.e., empty or full state) from an indication returned by a failed CSDBL. Failure of the CSDBL in Stone indicates that the return loop does not terminate.

Detecting Empty State

Stone discloses a Dequeue procedure that indicates an empty state, but the indication is distinct from the CSDBL. If the queue is empty when the Dequeue begins execution, then the CSDBL is never executed and a null value is returned from the Dequeue procedure. If an empty state occurs after the Dequeue procedure has entered the loop, then the loop terminates and the null value is returned. The Dequeue procedure detects the empty state by testing whether PrivateH is equal to 0, and not with a CSDBL. If an instance of the CSDBL in the Dequeue procedure fails, then the dequeue is retried. An empty state is not detectable from an indication returned by the CSDBL. Claims 1 and 25 variously recite returning from a DCAS, on failure of the DCAS, an indication of an empty state when the DCAS is executed as part of a pop operation. Independent claim 15 recites "interrogate instantaneous values of a first end index and a deque element adjacent to that identified thereby for a signature indicative of an empty state of the array." Stone discloses the CSDBL updating an end element of a queue and incrementing the corresponding index. Stone fails to disclose or suggest the CSDBL to interrogate as recited in claim 15.

As stated above, it is clear that the Office has misunderstood the disclosure of Stone. Stone fails to disclose or suggest any of the rejected claims.

Claim Rejections – 35 U.S.C. §103

Claims 6 – 14, 20 – 24, and 32 – 43 are rejected under 35 U.S.C. §103 as being unpatentable over Stone in view of view of Mark Allen Weiss’s Data Structures and Algorithm Analysis in C Second Edition, 1999 (hereinafter, “Weiss”). Claim 3 is rejected under 35 U.S.C. §103(a) as being unpatentable over Stone as applied to claim 1 above, and further in view of Weiss. Claim 29 is rejected under 35 U.S.C. §103(a) as being unpatentable over Stone as applied to claim 25 above, and further in view of Weiss.

Detecting Full State

The Examiner erroneously asserts that it would have been obvious to one of ordinary skill in the art to modify Stone with Weiss to achieve Applicant’s claims. The Examiner points to Weiss’ disclosure of checking for a full state. Weiss discloses checking for a full state with a conditional statement and an IsFull function. The IsFull function returns an indication of a full state as a Boolean value, not because of failure. Also, the IsFull function is not a DCAS. The use of an IsFull function in the Enqueue procedure of Stone, still fails to disclose or suggest returning an indication from failure of a DCAS whereby a full state is detectable as variously recited in claims 6, 33, 25, and 36.

Detecting Empty State

As stated above, Stone discloses a Dequeue procedure that indicates an empty state separate and distinct from the CSDBL. If the queue is empty when the Dequeue begins execution, then the CSDBL is never executed and a null value is returned from the Dequeue procedure. If an empty state occurs after the Dequeue procedure has entered the loop, then the loop terminates and the null value is returned. The CSDBL is not executed, does not fail, and does not return indication of an empty state upon failure. The Dequeue procedure detects the empty state by testing whether PrivateH is equal to 0, and not with a CSDBL. If one instance of the CSDBL in the Dequeue procedure fails, then the dequeue is retried. An empty state is not detectable from an indication returned by the CSDBL. Claims 34 recites “wherein, on failure, the DCAS returns an indication by which an empty state of the contiguous array is detected,” and

claim 36 recites “while returning on failure, an indication by which an empty state of the contiguous array is detected,” neither of which is disclosed or suggested by Stone.

Independent claim 20 recites “interrogate instantaneous values of a third end index and a deque element identified thereby for a signature indicative of a full state of the dequeue.” Stone does not disclose or suggest the CSDBL performing such interrogating for a signature indicative of a full state. Stone discloses the CSDBL updating an end element of a queue and incrementing the corresponding index.

Disambiguating Retry from a Boundary Condition

The Examiner contends that Stone discloses returning an indication disambiguating a retry state and a boundary condition state on failure of an access performed by DCAS. As stated above, indications returned by failed CSDBL in either one of the Dequeue and the Enqueue disclosed by Stone cause a looping and re-execution of the CSDBL and do not allow for detection of a boundary condition. Stone fails to disclose or suggest returning an indication from a failed DCAS disambiguating a retry state and a boundary condition state as variously recited in claims 10 and 43. Likewise, Stone also fails to disclose or suggest DCAS being responsive to a corresponding boundary condition state of the concurrent shared object as recited in claim 40.

Encoding a Distinguishing Value

The Office contends that Weiss discloses a circular buffer as recited in claim 32. Weiss discloses, at pages 111 – 112, a circular buffer with end indices front and back. However, assuming a circular array S, Weiss discloses enqueueing a value into the element S[back]. For a dequeue, the value at S[front] is returned. Weiss does not disclose or suggest encoding distinguishing values at the elements that correspond to the front and back. Hence, Weiss fails to disclose or suggest encoding distinguishing values at S[front] and at S[back] as recited in claim 32.

None of the art of record, standing alone or in combination, discloses or suggests any of the claims. Dependent claims are at least allowable because they depend from corresponding ones of the above allowable independent claims.

Conclusion

In summary, claims 1-43 are in the case. All claims are believed to be allowable over the art of record, and a Notice of Allowance to that effect is respectfully solicited. Nonetheless, if any issues remain that could be more efficiently handled by telephone, the Examiner is requested to call the undersigned at the number listed below.

CERTIFICATE OF MAILING OR TRANSMISSION

I hereby certify that, on the date shown below, this correspondence is being

- ☐ deposited with the US Postal Service with sufficient postage as first class mail and addressed as shown above.
☐ facsimile transmitted to the US Patent and Trademark Office.

Date

EXPRESS MAIL LABEL: _____

Respectfully submitted,



Steven R. Gilliam, Reg. No. 51,734
Attorney for Applicant(s)
(512) 338-6320 (direct)
(512) 338-6300 (main)
(512) 338-6301 (fax)